TCP wrapper based access List Rules can be included in the two files  
/etc/hosts.allow and  
/etc/hosts.deny .

Work precedence:

1. /etc/hosts.allow-
   1. if allow will not check 2
   2. if not found then go to 2
2. /etc/hosts.deny .
   1. if not found allow access.

The two tables that affect which clients can connect to which services (provided the service uses libwrap.a or is controlled by inetd/xinetd) are /etc/hosts.allow and /etc/hosts.deny. tcpd (tcp wrapper) uses /etc/hosts.allow and /etc/hosts.deny to perform its main task of checking for access control restrictions to system services.

Scenario 1: Delete /etc/hosts.allow . Add ALL: ALL to /etc/hosts.deny to deny all connections to the services (provided the service uses libwrap.a or is controlled by inetd/xinetd)

In this case, we set up an ftp server and this server is under control of xinetd

[root@localhost ~]# ftp 192.168.100.69

Connected to 192.168.100.69 (192.168.100.69).

421 Service not available, remote server has closed connection

Points to remember

* You can have only one rule per service in hosts.allow and hosts.deny file.
* Any changes to hosts.allow and hosts.deny file takesimmediate effect.
* The last line in the files hosts.allow and hosts.deny must be a new line character. Or else the rule will fail.

Rule Syntax  
The syntax for both hosts.allow and hosts.deny file takes the following form:

daemon : client [:option1:option2:…]

Examples

Allow SSH for xyz.com and deny access to all the others.

sshd : .xyz.com

… and in the hosts.deny file I include the rule:

sshd : ALL

Denys FTP access to all in abc.com domain  and hosts in the 192.168.1.0 network.

#FILE: /etc/hosts.deny  
vsftpd : 192.168.1. , .abc.com

Beautification

#FILE: /etc/hosts.deny  
vsftpd : 192.168.1. , .abc.co.in : spawn /bin/echo  `/bin/date` access denied >> /var/log/vsftpd.log : deny

In the above rule, spawn logs a message to the vsftpd log file each time the rule matches. deny is optional if you are including this rule in the hosts.deny file.

For example, you can use spawn option to send mail to the admin when ever a deny rule is matched.

Wildcards

You can use wildcards in the client section of the rule to broadly classify a set of hosts. These are the valid wildcards that can be used.

* ALL – Matches everything
* LOCAL – Matches any host that does not contain a dot (.) like localhost.
* KNOWN – Matches any host where the hostname and host addresses are known or where the user is known.
* UNKNOWN – Matches any host where the hostname or host address are unknown or where the user is unknown.
* PARANOID – Matches any host where the hostname does not match the host address.

**Patterns**

ALL : 123.12.

Matches all the hosts in the 123.12.0.0 network. Note the dot (.) in the end of the rule.

ALL : 192.168.0.1/255.255.255.0

IP address/Netmask can be used in the rule.

sshd : /etc/sshd.deny

If the client list begins with a slash (/), it is treated as a filename. In the above rule, TCP wrappers looks up the file sshd.deny for all SSH connections.

sshd : ALL EXCEPT 192.168.0.15

will allow ssh connection for only the machine with the IP address 192.168.0.15 and block all other connections.

You can use the options allow or deny to allow or restrict on a per client basis in either of the files hosts.allow and hosts.deny

in.telnetd : 192.168.5.5 : deny  
in.telnetd : 192.168.5.6 : allow

Shell Commands

As mentioned above, you can couple the rules to certain shell commands by using the following two options.

spawn – This option launches a shell command as a child process. For example, look at the following rule:

sshd : 192.168.5.5 : spawn /bin/echo `/bin/date` from %h >> /var/log/ssh.log : deny

Each time the rule is satisfied, the current date and the clients hostname %h is appended to the ssh.log file.

twist – This is an option which replaces the request with the specified command. For example, if you want to send to the client trying to connect using ssh to your machine, that they are prohibited from accessing SSH, you can use this option.

sshd : client1.xyz.com : twist /bin/echo “You are prohibited from accessing this service!!” : deny

When using spawn and twist, you can use a set of expressions. They are as follows :  
%a — The client’s IP address.  
%A — The server’s IP address.

%c — Supplies a variety of client information, such as the username and hostname, or the username and IP address.

%d — The daemon process name.  
%h — The client’s hostname (or IP address, if the hostname is unavailable).  
%H — The server’s hostname (or IP address, if the hostname is unavailable).

%n — The client’s hostname. If unavailable, unknown is printed. If the client’s hostname and host address do not match, paranoid is printed.

%N — The server’s hostname. If unavailable, unknown is printed. If the server’s hostname and host address do not match, paranoid is printed.

%p — The daemon process ID.

%s — Various types of server information, such as the daemon process and the host or IP address of the server.

%u — The client’s username. If unavailable, unknown is printed.